

stored at an address corresponding to the icons (first icons, second icons, third icons).

In another embodiment, when the process finds that the first icon is found in the associative database, it is determined if a pointer is stored with the first icon. When a pointer is not stored with the first icon, then a match has been found. When a pointer is stored with the first icon a next icon is determined. The next icon is the transform for the next non-overlapping window of the data block being searched. The next icon is compared to the an icon at the pointer location. When the next icon is the same as the icon at the pointer location a match has been found.

In another embodiment when the first icon is found in the associative database and includes a pointer, a second icon is determined. Next it is determined if the second icon has a matching the associative database. In another embodiment the second icon is determined using an icon append operation with a second portion to the first icon. The second portion is the next non-overlapping window of data in the data block being searched.

FIGs. 4 & 5 are a flow chart of the steps used in performing a sliding window search in accordance with another embodiment of the invention. The process starts, step 70, by generating an associative database at step 72. A first window of a data block is selected to be examined at step 74. The first window is iconized to form a first icon at step 76. A lookup in the associative database is performed to determine if there is a match at step 78. A second window of the data block is selected, wherein the second window contains a new portion and a

common portion of the first window at step 80. A second icon is determined using the first icon, a discarded portion and the portion but not the common portion at step 82. The second icon is associated with the second window which ends the process at step 84. In one
5 embodiment, this process is repeated until the complete data block has been examined. In another embodiment the process of forming an icon involves a linear feedback shift register operation. In another embodiment the linear feedback shift register operation is a cyclical redundancy code.

10 In another embodiment the process of forming the second icon includes determining a discarded icon for the discarded portion. Then an icon shift function is executed to form a shifted discarded icon. The shifted discarded icon is exclusive ORed with the first icon to form a seed icon. A new icon is determined for the new portion. The new icon is
15 exclusive ORed with the seed icon to form the second icon.

In another embodiment the lookup process to determine if there is a match includes determining if the associative database indicates a match, a no match or a qualifier match. When a qualifier match is indicated, a next window icon for the next complete non-overlapping
20 window of data is determined. Then it is determined if there is a pointer pointing from the first icon to the next window icon.

In another embodiment, when a qualifier match is indicated, a match length is determined. An extra portion is appended onto the first icon to form a second icon. Note the extra portion of the data plus the
25 window of data that has been iconized is equal to the match length.

Using the second icon it is determine if the associative database indicates a match.

FIG. 6 is a flow chart of the steps used in performing a sliding window search in accordance with another embodiment of the invention.

5 The process starts, step 90, by selecting a plurality of data strings to be found at step 92. The plurality of data strings are iconized to form a plurality of match icons at step 94. An associative database is created having a plurality of icons, wherein each of the match icons corresponds to one of the plurality of addresses at step 96. At step 98, a match flag is stored at each of the plurality of addresses corresponding to the
10 plurality of match icons which ends the process at step 100. When the plurality of data strings do not all have a same length a plurality of shortest data strings are selected. A plurality of short icons associated with the shortest data strings are determined. The match indicator is stored in the associative database at the address associated with each of
15 the short icons. A plurality of qualifier icons are determined for a first portion of a plurality of longer data strings. A qualifier flag is stored in the associative database for each of the qualifier icons. A match length indicator is stored with each of the qualifier icons in the associative
20 database. An icon is determined for a first window of a data block, wherein the first window has a window length equal to a shortest length. A lookup is performed in the associative database to determine if there is a match flag or a qualifier flag. When there is a qualifier flag, the match length indicator is retrieved. A complete icon is determined
25 for the portion of the data block equal to the match length. A lookup is